

## REMARKS

In the Office Action, the Examiner noted that Claims 1 through 10 were pending in the Application. The Examiner rejected all claims. Claims 1 - 10 have been amended and remain pending in this Application. Applicant traverses the rejections below.

### I. Oath/Declaration

The Office Action noted that the signatures of the inventors are missing. Applicants note that the signatures were provided in response to the Notice of Missing Parts, and the Declaration and Power of Attorney having the missing signatures was filed on April 21, 2001.

### II. Traversal of the Rejections under 35 USC Section 112

The Examiner rejected Claims 3 - 5, 9 and 10 under 35 USC Section 112, first paragraph.

Claims 3 - 5 were apparently rejected because “[t]he object can be accessed by another object.” This rational is not clear. This language does not appear in the claims. What object from the claims is being referred to is not specified or clear. The language used in the claims is used extensively throughout pages 4 - 10 of the Specification.

Generally, in object oriented programming, there is the concept that “another object” may contains “the object”'s reference, so the object can be accessed through “another object”.

This rejection is not clear, and Applicants are unable to respond without further details.

Claims 9 and 10 were rejected due to an objection to the Specification relative to the phrases “first subprocesses”, “second subprocesses”, “third subprocesses”, and “forth subprocesses”. There does not appear to be a rejection of the subject matter which follows these

terms. On pages 4 - 10 of the Specification, the language of the claims which follows these terms is certainly provided. While Applicants do not believe it makes any difference, these phrases have been amended in this Amendment to state "a process", which is certainly supported in the Specification.

The Examiner rejected Claim 8 under 35 USC Section 112, second paragraph, for lacking antecedent basis for the term "flag data". Applicants are not sure what this rejection is about. However, Claim 8 has been amended to make clear that the second occurrence of the term is preceded with the word "the".

### III. Traversal of the Rejections over the Cited Art

The Examiner rejected Claim 1 under 35 U.S.C. 102(e) as being anticipated by US Patent Number 6,209,066 to Holzle et al (Holzle). Claims 3, 4, 6, 7 and 9 were rejected as being anticipated by US Patent Number 5,862,376 to Steele, Jr. et al (Steele). Claim 2 was rejected as being unpatentable over Holzle in view of US Patent Number 5,652,883 to Adcock. Claims 5, 8 and 10 were rejected as being unpatentable over Steele in view of Adcock. Applicants traverse these rejections below.

#### A. The Present Invention

The present invention provides a technique for skipping a locking process for an object in memory when a thread accesses an object that only it will access in order to reduce the load imposed on a system and to improve the overall system performance. A program executing in a computer system has multiple threads that share and access objects stored in memory. The objects have thread locality flags associated therewith that indicate the presence or absence of thread localities. The threads examine the thread locality flags for the objects they attempt to access to determine whether the corresponding objects, which are to be accessed, have localities for the threads. If, so the, threads skip the locking process and access objects immediately. If

not, the object is locked prior to being accessed.

B. Differences between the Claims and the Cited Art

Claim 1 was rejected as being anticipated by Holzle.

Holzle discloses a technique that provides thread-local "allocation area" for "fast-allocating" threads and omits locking if the allocation area is thread-local. In contrast with the present invention, Holzle's "thread-local" flag will not be changed after the allocation area is prepared.

In contrast, the present invention handles thread locality of dynamically changing data (or objects). The flag may be dynamically turned off during the execution.

Another difference between the present invention and Holzle is that the present invention is mainly targeting objects allocated from the allocation area, and not targeting the allocation area itself. Accordingly, independent Claim 1 patentably distinguishes over Holzle. While it follows that dependent Claim 2 patentably distinguishes over the cited art, further differences will be provided below relative to Claim 2 and the cited art.

Claims 3, 4, 6, 7 and 9 are rejected as being anticipated by Steele.

Steele includes the idea of omitting the recursive locks. The Steele system omits lock processing if the object is already locked ("previously synchronized" in their words) by the thread. This omission is performed irrelevantly to whether the object is accessible from other threads or not.

The Steele "lock bit 226" does **not** mean that the object is inaccessible (unreachable) from other threads, so it is completely different from the thread-locality flag of the present invention.

Dependent Claim 2 is rejected over Holzle in view of Adcock.

Holzle discusses the locality of "allocation area", not the locality of each object. In contrast, the present invention describes the (dynamically changing) locality of each object allocated from the allocation area. Such a flag is not described in Holzle. Therefore, Applicant submits that dependent Claim 2 is not obvious at all and further distinguishes over the cited art..

Adcock shows an idea of generational garbage collection on a system with "conservative" stacks, which means that system cannot know whether each content of thread's stack is an object reference or not. The Adcock system needs to check stacks of all threads for garbage collection, as other previous garbage collections. In contrast, the present invention provides a new method for collecting memory area by just checking one thread's stack. Accordingly, Applicants submit that dependent Claim 2 further distinguishes over the cited art.

Claims 5, 8 and 10 are rejected over Steele and Adcock.

As described above, Steele's "lock bit" is completely different from our thread-locality flag. As described above, Adcock's garbage collector needs to check all threads' stacks. Therefore, the subject matter provided in the subject claims is not rendered obvious by the combination of Steele and Adcock. Accordingly, Claims 5, 8 and 10 also patentably distinguish over the cited art.

#### IV. Summary

Applicants have presented technical explanations and arguments fully supporting his position that the pending claims contain subject matter which is not taught, suggested or disclosed by cited art. Accordingly, Applicants submit that the present Application is in a condition for Allowance. Reconsideration of the claims and a Notice of Allowance are earnestly solicited.

Respectfully submitted,



Gregory M. Doudnikoff

Attorney for Applicant

Reg. No. 32,847

GMD:ssc

Docket No: JP919990286US1

PHONE: 919-254-1288

FAX: 919-254-4330

Serial No. 09/803,168

11